REPORT DOCUMENTATION PAGE

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE 18 Apr 97	3. REPORT TYPE AND DATES COVERED

4. TITLE AND SUBTITLE
Evaluation of Local Preconditioners for Multigrid Solutions of the Compressible Euler Equations

5. FUNDING NUMBERS

6. AUTHOR(S)
Barrett Taylor McCann

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Texas A&M University

8. PERFORMING ORGANIZATION REPORT NUMBER

97-018

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
DEPARTMENT OF THE AIR FORCE
AFIT/CI
2950 P STREET
WRIGHT-PATTERSON AFB OH 45433-7765

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A

Approved for public release;
Distribution Unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

14. SUBJECT TERMS

15. NUMBER OF PAGES
54

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT

19970424 031

EVALUATION OF LOCAL PRECONDITIONERS FOR MULTIGRID

SOLUTIONS OF THE COMPRESSIBLE EULER EQUATIONS

A Thesis

by

BARRETT TAYLOR MCCANN

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

December 1996

Major Subject: Aerospace Engineering

EVALUATION OF LOCAL PRECONDITIONERS FOR MULTIGRID

SOLUTIONS OF THE COMPRESSIBLE EULER EQUATIONS

A Thesis

by

BARRETT TAYLOR MCCANN

Submitted to Texas A&M University
in partial fulfillment of the requirements
for the degree of

MASTER OF SCIENCE

Approved as to style and content by:

_____
David L. Darmofal
(Chair of Committee)

_____
James H. Bramble
(Member)

_____
Leland A. Carlson
(Member)

_____
John L. Junkins
(Member)

_____
Donald T. Ward
(Interim Head of Department)

December 1996

Major Subject: Aerospace Engineering

ABSTRACT

Evaluation of Local Preconditioners for Multigrid

Solutions of the Compressible Euler Equations. (December 1996)

Barrett Taylor McCann, B.S., United States Air Force Academy

Chair of Advisory Committee: Dr. David L. Darmofal

The goal of this study is to examine and compare the effectiveness of two local preconditioners, when used in a multigrid algorithm, in accelerating the rate of convergence to an accurate steady solution of the two-dimensional compressible Euler equations. In this study, both the matrix preconditioner developed by Turkel and the block-Jacobi preconditioner are tested. While both preconditioners exhibit similar damping properties for error modes which are high-frequency in both coordinate directions (i.e., high-high modes), it is known that the Turkel preconditioner provides significantly better low-frequency propagation. In this thesis, this improved low-frequency propagation is shown to also improve (albeit nominally) the damping for modes which are high-frequency in only one direction (high-low and low-high modes). Thus, an important aspect of this work is assessing how improved low-frequency propagation can enhance multigrid convergence rates for preconditioned iterative techniques with similar damping properties. The results of first- and second-order numerical studies in a full-coarsening multigrid algorithm over several low freestream Mach numbers and with different boundary conditions indicate that the superior low-frequency propagation characteristics of Turkel's preconditioner result in better convergence rates than the block-Jacobi preconditioner. In addition, conclusions are drawn about the usefulness of multigrid with and without preconditioning, as well as the relative accuracy of the different solution methods used.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

FIGURE   Page

FIGURE

CHAPTER I

INTRODUCTION

In recent years, the rapid development of high-speed computers has led to increased use of computational techniques in predicting solutions to flows generated by specific geometric bodies. Certainly, the accuracy of computed solutions and realistic representations of such fluid dynamic phenomena as shocks and boundary layers are highly important considerations in designing and implementing a computational technique. However, though today's computers are admittedly far quicker and more powerful than those of even a decade ago, rapid rate of convergence is still a critical criterion in the selection of a computational fluid dynamics scheme. As ever more advanced schemes are more capable of analyzing flows about more complex geometries, it becomes critical to perform calculations upon very fine grids in areas where highly detailed and accurate solutions are required. As these complex schemes may easily require millions of calculations, taxing even the fastest and most powerful computers, it is necessary to implement techniques to minimize the number of calculations needed to generate an accurate steady solution.

To this end, acceleration techniques can be used which eliminate the time accuracy of the solution but allow faster convergence to the steady state (as long as the transient nature of the flow is not considered important). Two important techniques proposed in improving rate of convergence to a steady solution are multigrid and local preconditioning, both of which will be examined in this study.

A main feature of multigrid is the use of ever-coarser grids upon which a particular CFD scheme is applied [1]. The result of such a utilization is the representation

_____

The journal model is *AIAA Journal.*

of a low-frequency error mode as a higher-frequency mode on a coarser grid [2]. Iterative schemes, or smoothers, have been developed which rapidly damp high-frequency errors for the compressible Euler and Navier-Stokes equations [3, 4, 5, 6, 7]. A key ingredient of such smoothing methods is the use of a local block preconditioner to cluster high-frequency error modes. The use of multigrid in conjunction with a well-chosen smoother thus allows the quick high-frequency damping characteristics of the smoother to be applied to all error modes, substantially accelerating solution convergence [5, 7, 8].

A less understood phenomenon is the role of propagation in error removal for multigrid methods applied to hyperbolic problems. The pioneering work of Lötstedt [9] showed that low-frequency modes are accelerated by a factor of $\chi = (p+q)(2^l - 1)$ for p presmoothing iterations, q postsmoothing iterations, and l multigrid levels. Thus, assuming boundary conditions permit error modes to leave the domain, any techniques which accelerate low-frequency propagation should favorably affect multigrid convergence.

Preconditioners can be designed using either the original continuous equations or the discrete equations. The type of matrix preconditioners proposed by Turkel and van Leer are designed based on the continuous equations. These preconditioners have been designed to equalize different characteristic speeds which are present over the entire range of Mach numbers [10, 11]. Taking for example the 1-D Euler equations, one wave travels at a speed of $u + a$, another travels at u, and a third travels at $u - a$ for fluid speed u and speed of sound a. In one dimension, a local preconditioner can perfectly equalize these characteristic speeds [11]. In two dimensions, unfortunately, preconditioning is more difficult. In fact, no preconditioner can equalize the propagation speeds over all freestream Mach numbers; however, the van Leer and Turkel preconditioners optimally reduce the spread of the propagation speeds for the Euler

equations [12, 10]. In addition, these continuum-based preconditioners can have good error-damping characteristics [5].

A typical local matrix preconditioner based on the discrete equations is the block-Jacobi preconditioner. Jacobian-type preconditioners are based on the Jacobi iterative method; in this case the preconditioner is the block-diagonal matrix resulting from an approximate linearization of the discrete equations. The block-Jacobi preconditioner also displays improved error-damping characteristics [3, 4, 7] but, in contrast to the continuum-based preconditioners, does not have the improved propagation characteristics. Use of this type of preconditioner, therefore, may not provide convergence acceleration as effectively as the Turkel and van Leer preconditioners.

The goal of this study, then, is to evaluate the performance of the Turkel and block-Jacobi preconditioners in a full-coarsening multigrid algorithm. In particular, several comparisons will be made. First, the convergence rates of the preconditioned systems will be compared to the rate of convergence of the unpreconditioned system and to each other. Second, the dependence of each system on freestream Mach number will be examined. Third, the apparent accuracy of each system will be observed. Finally, conclusions will be drawn about the usefulness of multigrid techniques with and without preconditioning. The first- and second-order Euler solvers will be applied over a range of low freestream Mach numbers and to a variety of boundary conditions in order to extend observations to a wide range of low-speed test conditions.

In this thesis, Chapter II briefly describes the numerical methods used, including the Euler solver (Section A), the multigrid algorithm (Section B), and the two preconditioners (Section C). An analysis of the dissipation and propagation characteristics of the different systems is performed in Chapter III. Finally, in Chapter IV, the specific tests conducted are described, the results of these tests are shown, and conclusions are drawn about the performance of the numerical techniques employed.

CHAPTER II

NUMERICAL METHODS

The Euler equations comprise a hyperbolic system of inviscid fluid dynamic equations which represent the principles of mass, momentum, and energy conservation. This system of equations is useful when the viscous effects of a flow are negligible, such as when the boundary layer is very thin in comparison to the characteristic length of a flow field, and its interaction with the inviscid portion of the flow can thus be largely ignored. The usefulness of the Euler equations lies in the fact that they can be more efficiently solved numerically than the full Navier-Stokes equations [13].

The two-dimensional Euler equations are expressed in integral form as

$$\frac{d}{dt} \int U dA + \oint [F dy - G dx] = 0 \tag{1}$$

The four-element vector U contains mass, momentum, and energy variables:

$$U = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{bmatrix}$$

with local density ρ, x-velocity u, y-velocity v, and specific total energy E. The flux vectors F and G are defined as

$$F = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ \rho u H \end{bmatrix}$$

$$G = \begin{bmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ \rho v H \end{bmatrix}$$

where $p = (\gamma - 1)[\rho E - \frac{1}{2}\rho(u^2 + v^2)]$ is the local pressure and $H = \frac{a^2}{\gamma - 1} + \frac{1}{2}(u^2 + v^2)$ is the local stagnation enthalpy, with local speed of sound a and ratio of specific heats γ (which is here set to $\gamma = 1.4$, the specific heat ratio for air). The integrand dA is the differential area of the flowfield.

A.   Basic Solver

The numerical method used here is a structured-grid MUSCL scheme [14] using Roe's approximate Riemann solver [15]. This two-dimensional solution scheme is based on a one-dimensional approach, solving a 1-D Riemann problem at each cell face and combining the results from the four faces of each quadrilateral cell.

Using this approach, the Euler equations in each computational cell can be expressed in semi-discrete form as

$$A\frac{\partial U}{\partial t} + R = 0 \tag{2}$$

where the residual R represents the net flux in each cell and A is the cell area. Applying the MUSCL scheme, this residual is approximated as

$$R = \sum_{k=1}^{4} \mathbf{Q}_k \hat{F}_k \Delta s_k \tag{3}$$

where Δs is the length of a given cell face and the matrix $\mathbf{Q}$ rotates from an x-y

coordinate system to a grid-normal coordinate system:

$$\mathbf{Q} \equiv \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi & 0 \\ 0 & \sin\phi & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where ϕ is the grid-normal angle with respect to the x-axis. $\hat{F}$ is the flux at the cell face. Using the Roe scheme, this flux is given by

$$\hat{F} = \frac{1}{2}\left(F_{right} + F_{left}\right) - \frac{1}{2}\left|\hat{\mathbf{A}}\right|\left(U_{right} - U_{left}\right) \tag{4}$$

where $\hat{\mathbf{A}} = \frac{\partial F}{\partial U}$ and U and F are the grid-aligned state and flux vectors, respectively. The subscripts *left* and *right* indicate the states on either side of the cell face. The matrix $\left|\hat{\mathbf{A}}\right|$ has the same eigenvectors as $\hat{\mathbf{A}}$, but its eigenvalues are the absolute values of the eigenvalues of $\hat{\mathbf{A}}$. This flux approximation is more efficiently implemented as follows:

$$\hat{F} = \frac{1}{2}\left(F_{right} + F_{left}\right) - \frac{1}{2}\sum_{k=1}^{4}|\hat{\lambda}_k|\hat{v}_k\vec{r}_k \tag{5}$$

where $\hat{\lambda}_k$ are the eigenvalues of $\hat{\mathbf{A}}$, $\hat{v}_k$ are the dot-products of the left eigenvectors of $\hat{\mathbf{A}}$ and the state vector jumps $U_{right} - U_{left}$ (which represent the amplitudes of the jumps in the characteristics), and the vectors $\vec{r}_k$ are the right eigenvectors of $\hat{\mathbf{A}}$:

$$\lambda_1 = \hat{u} - \hat{a}$$

$$\lambda_{2,3} = \hat{u}$$

$$\lambda_4 = \hat{u} + \hat{a}$$

$$\hat{v}_1 = \frac{1}{2\hat{a}^2}[\Delta p - \hat{\rho}\hat{a}\Delta u]$$

$$\hat{v}_2 = \Delta\rho - \frac{1}{\hat{a}^2}\Delta p$$

$$\hat{v}_3 = \hat{\rho}\Delta v$$

$$\hat{v}_4 = \frac{1}{2\hat{a}^2}[\Delta p + \hat{\rho}\hat{a}\Delta u]$$

$$\vec{r}_1 = \begin{bmatrix} 1 \\ \hat{u} - \hat{a} \\ \hat{v} \\ \hat{H} - \hat{u}\hat{a} \end{bmatrix} \quad \vec{r}_2 = \begin{bmatrix} 1 \\ \hat{u} \\ \hat{v} \\ \frac{1}{2}(\hat{u}^2 + \hat{v}^2) \end{bmatrix} \quad \vec{r}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ \hat{v} \end{bmatrix} \quad \vec{r}_4 = \begin{bmatrix} 1 \\ \hat{u} + \hat{a} \\ \hat{v} \\ \hat{H} + \hat{u}\hat{a} \end{bmatrix}$$

with $\Delta() = ()_{right} - ()_{left}$. The terms $\hat{\rho}$, $\hat{u}$, $\hat{v}$, $\hat{H}$, and $\hat{a}$ are Roe-averaged values of density, x-velocity, y-velocity, total enthalpy, and speed of sound, respectively:

$$\sigma \equiv \sqrt{\rho_{right}/\rho_{left}}$$

$$\hat{\rho} = \sigma\rho_{left}$$

$$\hat{u} = \frac{u_{left} + \sigma u_{right}}{1 + \sigma}$$

$$\hat{v} = \frac{v_{left} + \sigma v_{right}}{1 + \sigma}$$

$$\hat{H} = \frac{H_{left} + \sigma H_{right}}{1 + \sigma}$$

$$\hat{a} = \sqrt{(\gamma - 1)[\hat{H} - \frac{1}{2}(\hat{u}^2 + \hat{v}^2)]}$$

For first-order solutions, the left and right state vectors, U_{left} and U_{right}, are the state vectors in the cells on either side of the face. In extending the scheme to second-order accuracy, van Leer's kappa scheme is used [14]. The kappa schemes make use of the state vectors in four adjacent cells instead of two. The left and right states at the jth cell face, U_j^+ and U_{j+1}^-, respectively, are defined as

$$U_j^+ \equiv U_j + \frac{1}{2}\delta U_j \tag{6}$$

and

$$U_{j+1}^- \equiv U_{j+1} - \frac{1}{2}\delta U_{j+1} \qquad (7)$$

where

$$\delta U_j \equiv \frac{1}{2}(1-\kappa)\phi_{j-1/2}^+(U_j - U_{j-1}) + \frac{1}{2}(1+\kappa)\phi_{j+1/2}^-(U_{j+1} - U_j)$$

$$\delta U_{j+1} \equiv \frac{1}{2}(1-\kappa)\phi_{j+3/2}^-(U_{j+2} - U_{j+1}) + \frac{1}{2}(1+\kappa)\phi_{j+1/2}^+(U_{j+1} - U_j)$$

The limiters ϕ are designed to preserve monotonicity and to ensure the scheme is total variation diminishing. In this work, $\kappa = 0$ is used; this is Fromm's central difference gradient. In addition, no limiter is used (i.e. $\phi = 1$), since smooth flows at low Mach numbers do not exhibit discontinuities and limiting is therefore unnecessary. In this implementation, the primitive state vector is used to perform the higher-order interpolation:

$$U_p = \begin{bmatrix} \rho \\ u \\ v \\ p \end{bmatrix}$$

The implementation of the MUSCL scheme, therefore, involves calculating the flux at each cell face, combining these fluxes over the four faces of each cell, and then computing each cell update. In this work, a multistage optimal damping scheme is used:

$$U^{(0)} = U^n$$

$$U^{(1)} = U^{(0)} - \alpha_1 \frac{\nu \Delta t_{max}}{A} R(U^{(0)}) \qquad (8)$$

$$U^{(2)} = U^{(0)} - \alpha_2 \frac{\nu \Delta t_{max}}{A} R(U^{(1)})$$

$$U^{n+1} = U^{(2)}$$

where ν is the Courant number. For this unpreconditioned iterative scheme, Δt_{max}

is defined as

$$\Delta t_{max}^{-1} = \frac{\tau}{A} \sum_{k=1}^{4} \lambda_{max,k} \Delta s_k \tag{9}$$

with $\tau = 1$ for first-order accuracy and $\tau = 1 - \kappa$ for second-order accuracy; λ_{max} is the largest eigenvalue. For first-order solutions, a single-stage scheme is used (i.e. $\alpha_1 = 0$, $\alpha_2 = 1$) with $\nu = 1$. For the second-order scheme, the coefficients α_1 and α_2 are here selected to be 0.4978 and 1.0, respectively; the Courant number $\nu = 1.7598$ is also used. These values are shown to be optimal for a two-stage $\kappa = 0$ scheme by Lynn [5]. (It is noted that this optimal Courant number differs from Lynn's by a factor of two, due to a slight difference in the definition of the Courant number used here.)

B. Multigrid

Multigrid solution algorithms utilize multiple scales of discretization, allowing the damping properties of a well-chosen smoother to be applied to all error modes in an approximation to the steady solution. The process for achieving this goal is briefly outlined as follows. First, represent an initial solution on the desired fine grid. Apply an iterative scheme until the high-frequency components of the error are eliminated and only smooth, low-frequency error modes remain. Next, interpolate this new solution to a coarser grid (with each coarse grid cell composed of four fine grid cells, for example) via a restriction operator. Doing so will represent the low-frequency error modes from the fine grid solution as higher-frequency modes on the coarser grid. Again, apply the iterative scheme to damp these high-frequency error components, and restrict to yet a coarser grid. Repeat this process until the solution is smoothed on the coarsest desired grid. Then, through a prolongation process, extrapolate this coarse grid solution back to each successive finer grid (smoothing again on each grid,

if desired) until a new solution is obtained on the original finest grid. Assuming, therefore, that a smoother can be developed which efficiently damps high-frequency error components on a fine grid, allowing only low-frequency modes to be passed to the next coarser grid, multigrid can then accelerate convergence to a steady solution by applying the good smoothing properties of an iterative scheme to a much larger range of error components than the same smoother could efficiently affect on a single grid.

This multigrid process can be written in recursive form as $v^h \leftarrow NMV^h(v^h, f^H)$ and is described by Lynn [5] as follows. Consider a nonlinear problem to be solved on a fine grid Ω^h:

$$N^h(u^h) = f^h \tag{10}$$

If an approximate solution for u^h on this grid is v^h, then the residual can be expressed as

$$r^h = f^h - N^h(v^h) \tag{11}$$

A restriction operator, which interpolates an approximation from a fine grid Ω^h to the next coarser grid Ω^{2h}, is I_h^{2h}, and a prolongation operator, which returns a coarse grid approximation to the next finer grid, is I_{2h}^h. The algorithm $v^h \leftarrow NMV^h(v^h, f^H)$ is:

- Using smoother P, relax p times on Ω^h to obtain an approximate solution v^h

- If Ω^h is not the coarsest grid:

  - $r^h \leftarrow f^h - N^h(v^h)$

  - $f^{2h} \leftarrow I_h^{2h} r^h + N^{2h}(I_h^{2h} v^h)$

  - $v^{2h} \leftarrow I_h^{2h} v^h$

  - $v^{2h} \leftarrow NMV^{2h}(v^{2h}, f^{2h})$

– Correct $v^h \leftarrow I_{2h}^h(v^{2h} - I_h^{2h}v^h)$

- Using smoother P, relax q times on Ω^h to improve the approximate solution v^h

The algorithm NMV as described here follows the full-coarsening algorithm. This algorithm, by simultaneously coarsening in both the x- and y-directions, requires the smoother to damp modes which are high-frequency in any direction. Unfortunately, the explicit multistage scheme can only effectively smooth errors which are high-frequency in both directions (high-high modes). This means that the error modes which are high-frequency in one coordinate direction but low-frequency in the other (high-low and low-high) will not be efficiently damped by the iterative scheme. Use of a semi-coarsening multigrid algorithm, however, can overcome this limitation by coarsening in only one coordinate direction at a time, therefore representing high-low and low-high modes as high-high on their respective coarser grids [16]. This algorithm adds additional complexity to multigrid implementation and was not used in this study, but remains for future investigations.

In practice, it is not necessary to use a recursive algorithm to implement a multi-grid cycle. Multigrid can be more conveniently implemented as follows:

- Using a modification of the multistage solver defined in the previous section, relax p times on the finest grid:

$$U_h^{(0)} = U_h^n$$

$$U_h^{(1)} = U_h^{(0)} - \alpha_1 \frac{\nu \Delta t_{h,max}}{A_h}[R(U_h^{(0)}) + P_h] \tag{12}$$

$$U_h^{(2)} = U_h^{(0)} - \alpha_2 \frac{\nu \Delta t_{h,max}}{A_h}[R(U_h^{(1)}) + P_h]$$

$$U_h^{n+1} = U_h^{(2)}$$

where the multigrid source term $P_h = 0$ on the finest grid Ω^h and is here defined

on coarser grids as $P_{2h} = \sum_{k=1}^{4}[R(U_h) + P_h] - R(U_{2h})$

- Restrict the new fine grid solution U_h to the next coarser grid: $U_{2h} = I_h^{2h}(U_h)$

- Using the multistage solver, relax p times on the coarse grid

- Repeat until the solution is relaxed p times on the coarsest grid

- Using the multistage solver, relax q times on the coarse grid

- Prolongate the new coarse grid solution U_{2h} to the next finer grid: $U_h = U_h + I_{2h}^h(U_{2h} - I_h^{2h}U_h)$

- Using the multistage solver, relax q times on the fine grid

- Repeat until the solution is relaxed q times on the finest grid

In this study, two presmoothing updates and two postsmoothing updates were performed on each grid level (i.e., $p = 2$ and $q = 2$), making the multigrid algorithm used a V-cycle. The restriction operator was a four-cell area-weighted average:

$$U_{2h}A_{2h} = \sum_{n=1}^{4} U_h A_h \tag{13}$$

The prolongation operator was as described by Tai [8], using the three coarse grid cells nearest to the cell-center of the fine grid cell:

$$U_h = U_h + \frac{1}{2}(\Delta U_{2h})_1 + \frac{1}{4}(\Delta U_{2h})_2 + \frac{1}{4}(\Delta U_{2h})_3 \tag{14}$$

where the fine grid cell is included in coarse grid cell 1 and coarse grid cells 2 and 3 are its neighbors, as shown in Figure 1. The terms ΔU_{2h} are the changes in the state values on the coarse grid from before the prolongation to after the prolongation.
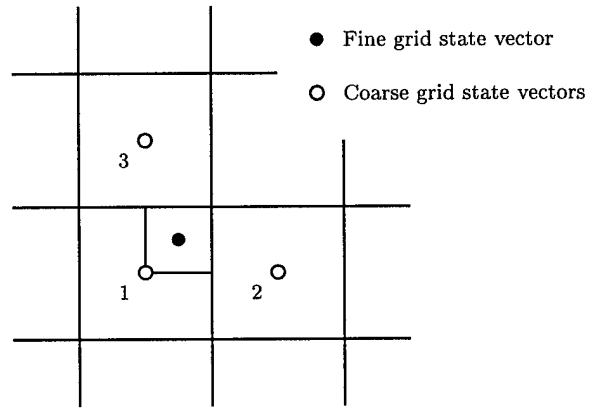
Fig. 1. Multigrid prolongation operator.

C. Local Preconditioning

The continuum-based Turkel preconditioner and the discrete-based block-Jacobi preconditioner are now discussed. Implementation of these two preconditioners will first be described; a comparative analysis of their dissipation and propagation characteristics will then be performed.

The Euler equations can be expressed in quasi-linear form as

$$\frac{\partial U}{\partial t} + \mathbf{A}\frac{\partial U}{\partial x} + \mathbf{B}\frac{\partial U}{\partial y} = 0 \tag{15}$$

where $\mathbf{A} = \frac{\partial F}{\partial U}$ and $\mathbf{B} = \frac{\partial G}{\partial U}$. Preconditioner analysis and implementation is simplified via use of the symmetrized variables. The symmetrized variable system is defined as

$d\tilde{U} \equiv \mathbf{M}dU$, where

$$dU = \begin{bmatrix} \frac{dp}{\rho a} \\ du \\ dv \\ ds \end{bmatrix}$$

and

$$\frac{\partial \tilde{U}}{\partial t} + \tilde{\mathbf{A}}\frac{\partial \tilde{U}}{\partial x} + \tilde{\mathbf{B}}\frac{\partial \tilde{U}}{\partial y} = 0 \tag{16}$$

with

$$\tilde{\mathbf{A}} = \begin{bmatrix} u & a & 0 & 0 \\ a & u & 0 & 0 \\ 0 & 0 & u & 0 \\ 0 & 0 & 0 & u \end{bmatrix} \qquad \tilde{\mathbf{B}} = \begin{bmatrix} v & 0 & a & 0 \\ 0 & v & 0 & 0 \\ a & 0 & v & 0 \\ 0 & 0 & 0 & v \end{bmatrix}$$

ds is the linearized entropy perturbation. In transforming from the symmetrized variables to the conserved variables, transformation matrices $\mathbf{M}$ and $\mathbf{M}^{-1}$ are used such that $\tilde{\mathbf{A}} = \mathbf{M}^{-1}\mathbf{A}\mathbf{M}$ and $\tilde{\mathbf{B}} = \mathbf{M}^{-1}\mathbf{B}\mathbf{M}$. These transformation matrices are defined in [5] as

$$\mathbf{M} \equiv \begin{bmatrix} \frac{\rho}{a} & 0 & 0 & -\frac{1}{a^2} \\ \frac{\rho u}{a} & \rho & 0 & -\frac{u}{a^2} \\ \frac{\rho v}{a} & 0 & \rho & -\frac{v}{a^2} \\ \rho a\left(\frac{1}{2}M^2 + \frac{1}{\gamma-1}\right) & \rho u & \rho v & -\frac{1}{2}M^2 \end{bmatrix}$$

and

$$\mathbf{M}^{-1} \equiv \begin{bmatrix} \frac{\gamma-1}{2}\frac{aM^2}{\rho} & -(\gamma-1)\frac{u}{\rho a} & -(\gamma-1)\frac{v}{\rho a} & \frac{\gamma-1}{\rho a} \\ -\frac{u}{\rho} & \frac{1}{\rho} & 0 & 0 \\ -\frac{v}{\rho} & 0 & \frac{1}{\rho} & 0 \\ a^2(\frac{\gamma-1}{2}M^2 - 1) & -(\gamma-1)u & -(\gamma-1)v & \gamma-1 \end{bmatrix}$$

where $M = \frac{\sqrt{u^2+v^2}}{a}$ is the local Mach number.

Turkel's local preconditioner, which is valid only for low Mach number flows, is applied to the two-dimensional Euler equations as follows:

$$A\frac{\partial U}{\partial t} + \mathbf{P}R = 0 \tag{17}$$

where $\mathbf{P}$ is the preconditioner matrix. In terms of symmetrized variables, with $\mathbf{P} = \mathbf{M}\tilde{\mathbf{P}}\mathbf{M}^{-1}$, the Turkel preconditioner is defined as

$$\tilde{\mathbf{P}} = \begin{bmatrix} \beta^2 & 0 & 0 & 0 \\ -\frac{u}{a}\left(1+\beta^2\right) & 1 & 0 & 0 \\ -\frac{v}{a}\left(1+\beta^2\right) & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{18}$$

with x-velocity u, y-velocity v, speed of sound a, and, with local Mach number M and freestream Mach number M_∞, $\beta \equiv \max\left(M, \eta M_\infty\right)$ for free parameter $0 \le \eta \le 1$. This free parameter is set to $\eta = 0.5$ in this work and is necessary to limit transient growth due to eigenvector non-orthogonality as $M \to 0$ [17].

Within the framework of Roe's approximate Riemann solver and a multigrid algorithm, Turkel's preconditioner is implemented via the following two modifications to the basic solver. First, in calculating the residual, the flux function is altered such that

$$\hat{F} = \frac{1}{2}\left(F_{right} + F_{left}\right) - \frac{1}{2}\mathbf{M}\hat{\mathbf{P}}^{-1}\left|\hat{\mathbf{P}}\hat{\mathbf{A}}\right|\mathbf{M}^{-1}\left(U_{right} - U_{left}\right) \tag{19}$$

This formulation ensures that as the steady state is approached, the preconditioned

system approximates the original system. The expression $\hat{\mathbf{P}}^{-1} \left| \hat{\mathbf{P}} \hat{\mathbf{A}} \right|$ is provided:

$$\hat{\mathbf{P}}^{-1} \left| \hat{\mathbf{P}} \hat{\mathbf{A}} \right| = \begin{bmatrix} \frac{\sqrt{a^2 - u^2}}{\beta} & 0 & 0 & 0 \\ \frac{u(1+\beta)\sqrt{a^2 - u^2}}{\beta a} & \beta \sqrt{a^2 - u^2} & 0 & 0 \\ D_{31} & D_{32} & |u| & 0 \\ 0 & 0 & 0 & |u| \end{bmatrix} \tag{20}$$

with

$$D_{31} = \frac{v(1+\beta)\left[u^2(1+\beta^2)\sqrt{a^2 - u^2} - |u|\, a^2 \beta\right]}{\beta a \left(u^2 + u^2 \beta^2 - a^2 \beta^2\right)}$$

$$D_{32} = \frac{uv(1+\beta)\left[\beta\sqrt{a^2 - u^2} - |u|\right]}{u^2 + u^2 \beta^2 - a^2 \beta^2}$$

The Roe-averaged values for u, v, and a are used in this flux calculation. Efficient implementation of this flux approximation makes use of the fact that $\mathbf{M}^{-1}(U_{right} - U_{left}) = \tilde{U}_{right} - \tilde{U}_{left}$ and involves evaluating $\mathbf{M}\hat{\mathbf{P}}^{-1}|\hat{\mathbf{P}}\hat{\mathbf{A}}|(\tilde{U}_{right} - \tilde{U}_{left})$ via matrix-vector multiplication instead of computationally expensive matrix-matrix multiplication. This modification to the flux function has been previously shown to increase low Mach number accuracy [18, 19, 20].

Second, the state vector update for each cell is modified such that, for example,

$$U^{(n+1)} = U^{(n)} - \nu \frac{\Delta t_{max}}{A} \mathbf{P}\left(R + P_{2h}\right) \tag{21}$$

It is noted that the computation of the local timestep is now based on the eigenvalues of $\mathbf{PA}$:

$$\lambda_1 = u$$

$$\lambda_2 = u$$

$$\lambda_3 = \beta \sqrt{a^2 - u^2}$$

$$\lambda_4 = -\beta \sqrt{a^2 - u^2}$$

The development of the Turkel preconditioner is described in detail in [21, 22].

In addition, the block-Jacobi preconditioner was applied to this multigrid algorithm. In terms of conserved variables, the inverse of the preconditioner, $\mathbf{P}^{-1}$, for a particular cell is defined as a sum over the four cell faces:

$$\mathbf{P}^{-1} = \frac{\tau \Delta t_{max}}{A} \sum_{k=1}^{4} \mathbf{Q}_k^{-1} \left| \hat{\mathbf{A}}_k \right| \mathbf{Q}_k \Delta s_k \qquad (22)$$

where Δs is the cell face length and $\left| \hat{\mathbf{A}}_k \right|$ is based on the Roe-averaged grid-normal state vector. Based on this definition, it is clear that the four-by-four matrix $\mathbf{P}^{-1}$ must be numerically inverted in order to obtain the preconditioner matrix $\mathbf{P}$. This is done most efficiently by solving for the preconditioned residual $\mathbf{P}R$ directly, rather than by finding $\mathbf{P}$ and then applying the preconditioner to the residual R. It is noted that, because of the inclusion of $A/\Delta t_{max}$ in the block-Jacobi preconditioner, the update becomes independent of a timestep, which thus does not need to be calculated. In addition, the flux function does not require modification, as is required by Turkel's preconditioner. The reader is directed to [3, 7] for more detail about the application of the block-Jacobi preconditioner to the 2-D compressible Euler and Navier-Stokes equations.

The high-frequency damping and low-frequency propagation characteristics of the unpreconditioned Euler equations and the Turkel and block-Jacobi preconditioned equations are now compared. To determine the high-frequency damping, the discrete system is linearized about a constant mean state and a Fourier decomposition in the x- and y-directions is assumed [3]. This gives two grid wave numbers, θ_x and θ_y, representing the x- and y-directions, respectively, which vary from $-\pi$ to π.

A high frequency mode is generally defined as $\pi/2 < |\theta| \leq \pi$. For full-coarsening multigrid, the appropriate frequencies which must be damped by the smoother are high-low, high-high, and low-high in the x- and y-directions. For semi-coarsening

multigrid, only the high-high frequencies must be damped [4].

Figures 2-4 depict the low-low, high-low, low-high, and high-high frequency content of the three second-order systems, superimposed on the amplification factors for the optimal two-stage scheme described earlier. The flow is oriented at an angle $\alpha = 20°$ from the x-axis. Only the analyses with a freestream Mach number of $M_\infty = 0.1$ are shown. However, similar patterns are evident for other low Mach number flows.

As can be seen in Figure 2, at this freestream Mach number, the Euler equations contain several high-frequency error components quite close to the $g = 1$ amplification contour, resulting in very poor damping of these modes. This is true not only for the high-low and low-high errors (which could be overcome by a semi-coarsening algorithm), but also for high-high error modes. This fact indicates that no multigrid algorithm will likely be able to quickly damp all high-frequency components. Based on this result, it is theorized that an unpreconditioned system will not largely realize the benefits of multigrid.

By contrast, the Turkel and block-Jacobi preconditioners (Figures 3 and 4, respectively) produce maximum amplification factors of high-high error modes of approximately 0.72 (Turkel) and 0.79 (block-Jacobi). With this result in mind, it is likely that, under conditions where only dissipation could affect error elimination (a domain with periodic boundaries, for example), the convergence rates of the two preconditioned systems would be quite good when a semi-coarsening algorithm is used and would compare favorably with each other. A look at the high-low and low-high damping, however, shows that the two systems would likely not perform as well in a full-coarsening scheme. The Turkel preconditioned system still exhibits good high-low damping ($g_{max} \approx 0.76$), but low-high damping suffers ($g_{max} \approx 0.96$). The block-Jacobi system exhibits poor damping of both high-low and low-high error
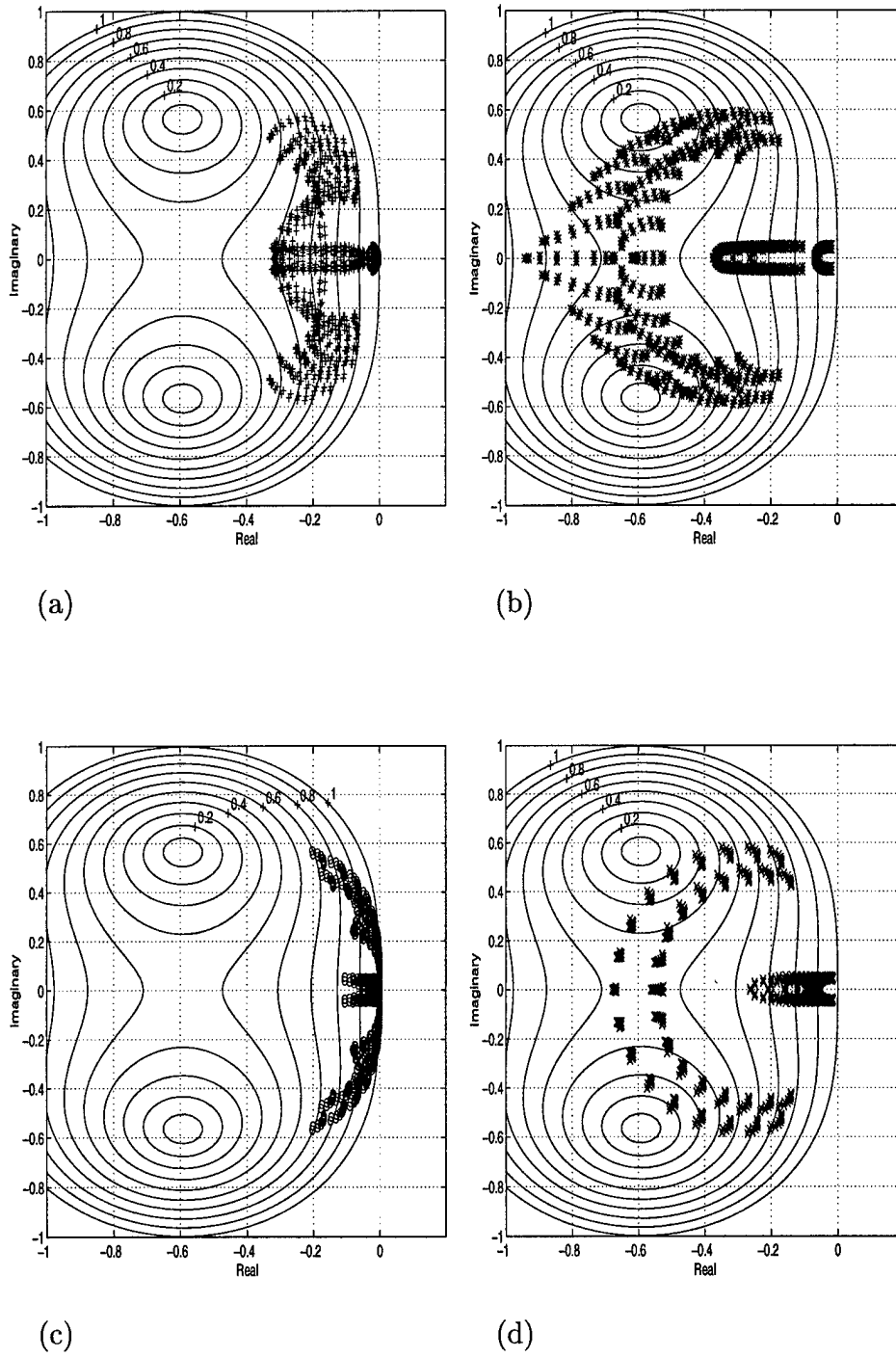
Fig. 2. Fourier footprint, Euler equations, 64x32 square grid, $M_\infty = 0.1$, $\alpha = 20°$, optimal two-stage iteration, amplification contours superimposed: a) low-high components, b) high-high components, c) low-low components, and d) high-low components.
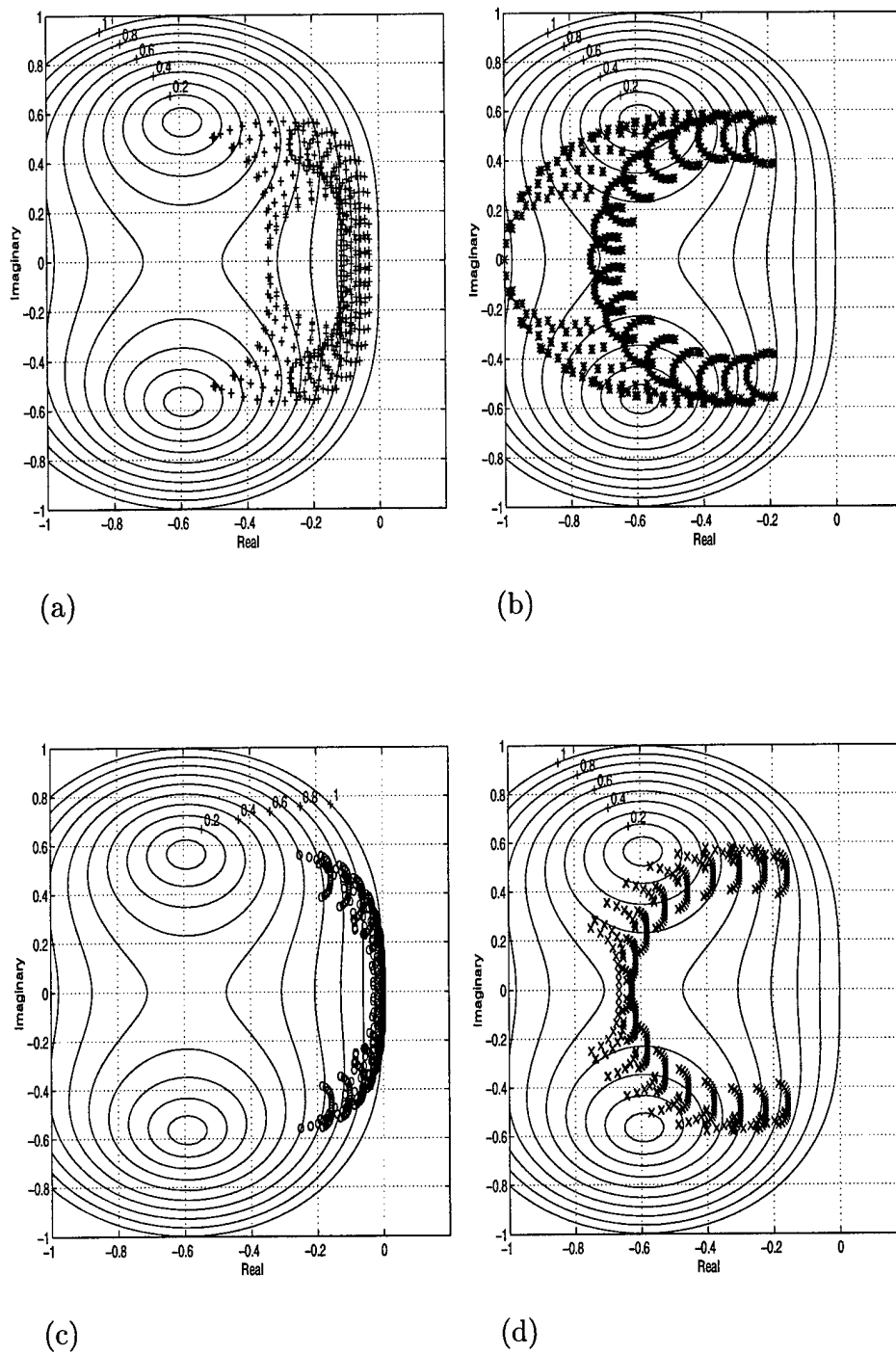
Fig. 3. Fourier footprint, Turkel preconditioner, 64x32 square grid, $M_\infty = 0.1$, $\alpha = 20°$, optimal two-stage iteration, amplification contours superimposed: a) low-high components, b) high-high components, c) low-low components, and d) high-low components.

Fig. 4. Fourier footprint, block-Jacobi preconditioner, 64x32 square grid, $M_\infty = 0.1$, $\alpha = 20°$, optimal two-stage iteration, amplification contours superimposed: a) low-high components, b) high-high components, c) low-low components, and d) high-low components.
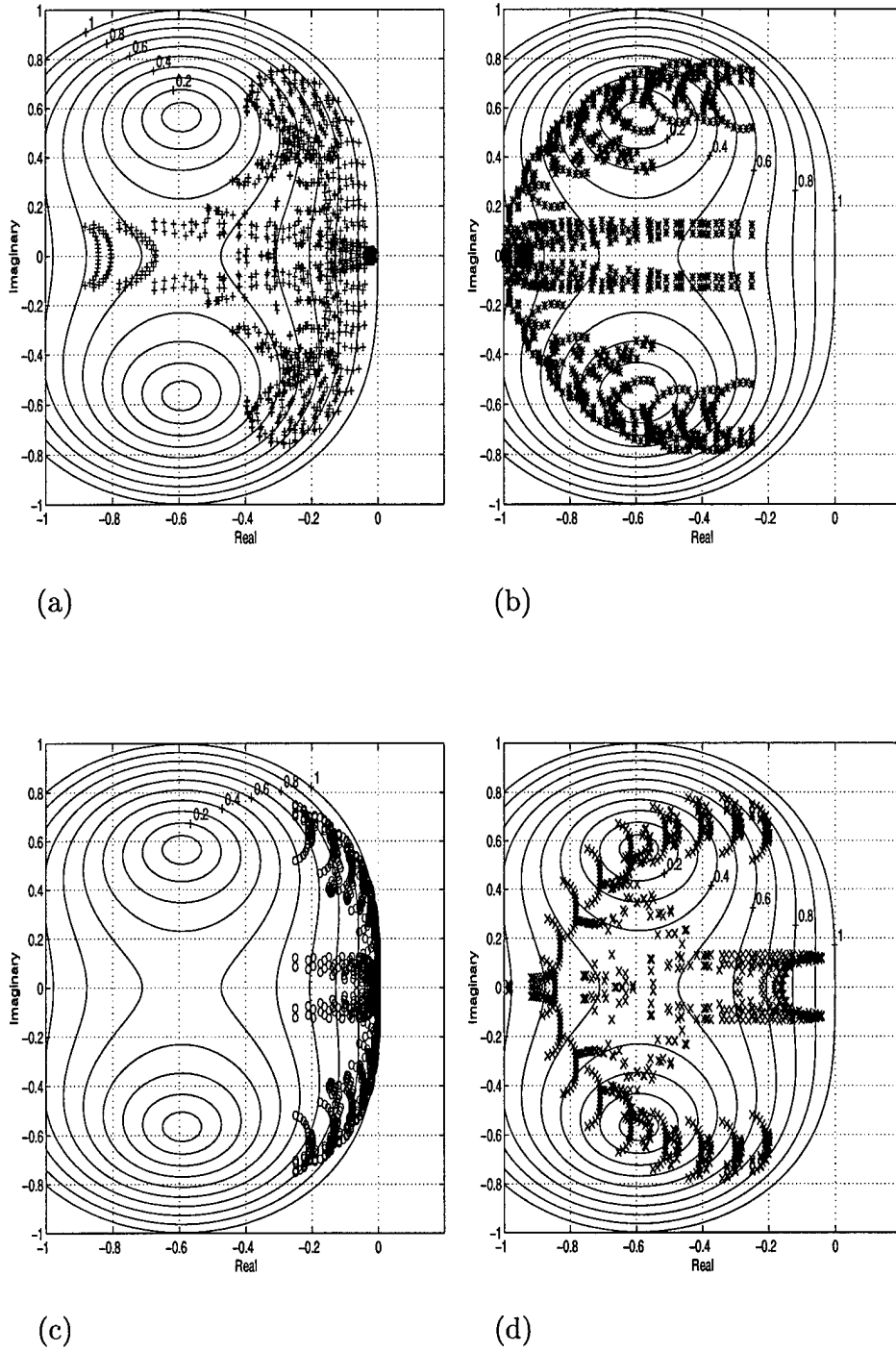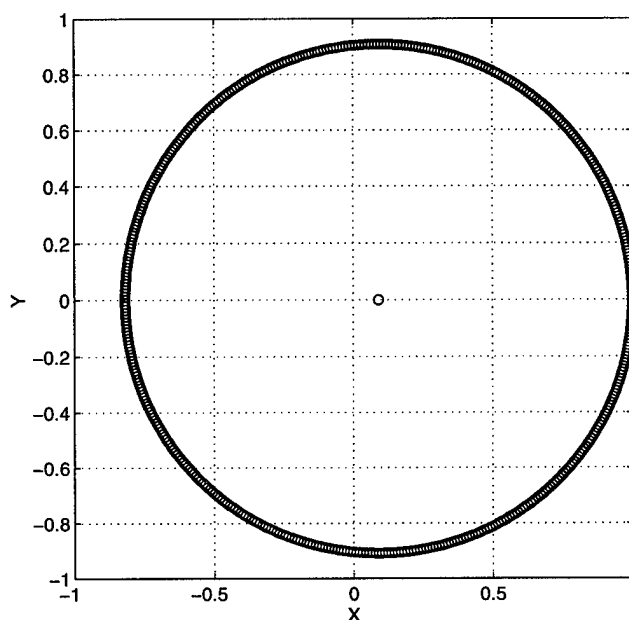
Fig. 5. Wave fronts, Euler equations, $M_\infty = 0.1$.

components ($g_{max} \approx 0.94$ and $g_{max} \approx 0.98$, respectively). Thus, it is apparent that, if incorporated into a full-coarsening multigrid algorithm, the two preconditioners will not perform optimally (with the Turkel preconditioner perhaps somewhat better than the block-Jacobi).

To examine the low-frequency propagation characteristics of the three systems, the wave fronts resulting from a point disturbance are determined, following the work in [10]. Figures 5-7 show these wave fronts for the three systems at $M_\infty = 0.1$. The plots have been appropriately scaled such that the largest distance traveled is one.

The Euler equations have two modes, entropy and vorticity, which convect with the freestream velocity and, at this low Mach number, are located near the origin. By contrast, the acoustic waves convect with the freestream velocity while simultaneously radiating outward at the speed of sound. This results in widely varying propagation speeds as $M_\infty \to 0$.

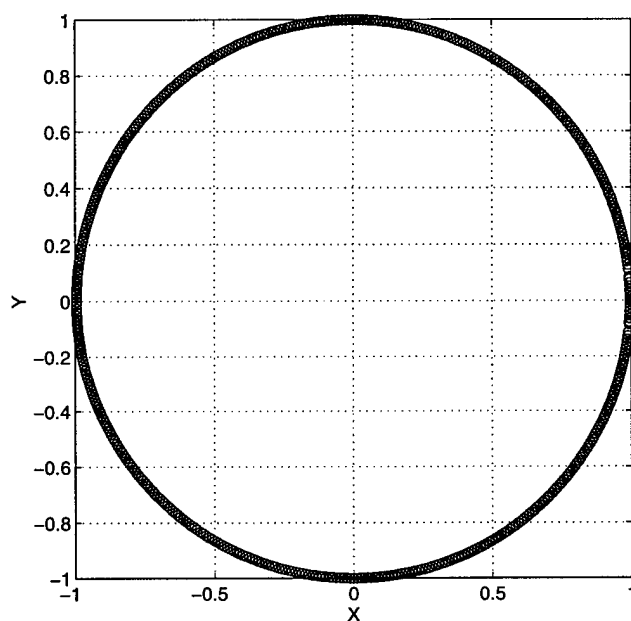The Turkel preconditioner successfully removes this variation in propagation

Fig. 6. Wave fronts, Turkel preconditioner, $M_\infty = 0.1$.

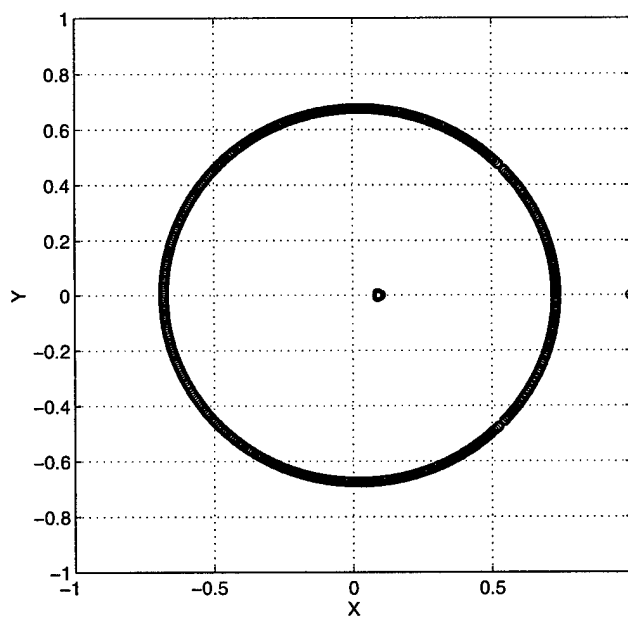Fig. 7. Wave fronts, block-Jacobi preconditioner, $M_\infty = 0.1$.

speeds. Although not evident in Figure 6, the convective modes are now located at (1,0), traveling with the same speed as the acoustic waves.

The block-Jacobi preconditioner places one convective mode (entropy) at (1,0), while the other convective mode (vorticity) remains close to the origin. The acoustic modes are also evident and are approximately located on a circle with radius of about 0.68, centered just to the right of the origin. It is the presence of the low-speed vorticity mode, which eventually falls on the origin as $M_\infty \rightarrow 0$, that results in the poor propagation characteristics of this preconditioner.

These wave front plots, then, suggest that the unpreconditioned and block-Jacobi preconditioned systems will not be able to efficiently propagate all low-frequency error modes out of a domain. Because the Turkel preconditioned system has all its smooth wave speeds equalized, however, all low-frequency error modes will be allowed to propagate out of the domain with equal efficiency. Thus, it is suspected that a flow simulation with free boundary conditions, which allow errors to propagate away, will benefit from this wave equalization characteristic of the Turkel preconditioner, and this preconditioned system will accelerate convergence more rapidly than the other two systems.

CHAPTER III

NUMERICAL STUDIES

Several numerical tests were performed to compare the effectiveness of the Turkel and block-Jacobi local preconditioners. First, random perturbations to the freestream values were applied over a 64x32 grid on a square domain. This simple test would be used to illustrate the benefits of using the two preconditioned systems in place of the system without preconditioning. In addition, the use of free boundaries on the four sides of the domain would help to illustrate the importance of a preconditioner's propagation characteristics in improving convergence rates; identical tests with periodic boundaries would illustrate the relative dissipation properties of the unpreconditioned system and the two preconditioned systems. Second, flow over a solid circular arc bump was simulated. This flow was selected in order to evaluate the utility of the preconditioners in studying a more realistic problem. All tests were repeated for three low freestream Mach numbers ($M_\infty = 0.05$, $M_\infty = 0.1$, $M_\infty = 0.2$). Both first- and second-order solvers were used in each test. The use of single grid and two-, three-, and four-grid algorithms were used in each instance. The variety of tests were conducted with different solvers at different Mach numbers in order draw broader conclusions with applicability to a wide range of CFD applications.

Comparisons of convergence rates are often based on the work unit, which is defined here as the amount of work required to calculate a residual on the finest grid. The work unit for a first-order test, then, is computationally smaller than a second-order work unit. In this study, work is considered to be based solely on the work required to calculate the residuals on each successive grid level within the multigrid algorithm. The work added due to intergrid transfers (prolongation and restriction) and to the application of local preconditioning is ignored, although these operations

generally add a ten to fifteen percent increase in actual work, with the block-Jacobi preconditioner being more expensive than the Turkel preconditioner.

A.   Preconditioning Effectiveness

For the first test case (free boundaries), a random small perturbation (between -0.01% and +0.01%) was added to the freestream condition. To eliminate grid-alignment-related problems, a flow angle of 20° to the x-axis was imposed. Optimal Courant numbers and multistage coefficients were used as mentioned in the previous section.

Tests were run with both first- and second-order solvers over the three Mach numbers and four multigrid algorithms. In all cases, a V-cycle was used with two presmoothing iterations ($p = 2$) and two postsmoothing iterations ($q = 2$) performed on each grid level. Solutions were considered to be converged when the root-mean-square of the residual was decreased by six orders of magnitude.

The amount of work required to achieve the desired level of convergence for each first-order test case is shown in Table 1. The work requirements for convergence for the second-order cases are listed in Table 2. The work requirements listed show that, for both first- and second-order solvers, over all three Mach numbers, both preconditioners provided significant improvements in convergence rates over the unpreconditioned system (as expected). However, though it is clear that either preconditioner is useful, it is apparent that the Turkel preconditioner performs noticeably better than the block-Jacobi preconditioner. As mentioned in the previous section, the slightly better high-low damping and vastly better propagation characteristics of the Turkel preconditioner would seem to explain this advantage.

Table 1. Work Requirements for Convergence, First-Order Solver, 64x32 Square Grid, Free Boundaries, $\alpha = 20°$, $p = 2$, $q = 2$

	$M_\infty = 0.05$		
Grids	No PC	Turkel PC	Block-Jacobi PC
1	5373	753	917
2	3414	439	751
3	2890	280	805
4	2921	224	815
	$M_\infty = 0.1$		
Grids	No PC	Turkel PC	Block-Jacobi PC
1	2865	757	885
2	2020	439	589
3	1643	280	566
4	1656	224	578
	$M_\infty = 0.2$		
Grids	No PC	Turkel PC	Block-Jacobi PC
1	1573	761	749
2	1176	445	457
3	948	294	417
4	961	224	425

Table 2. Work Requirements for Convergence, $\kappa = 0$ Solver, 64x32 Square Grid, Free Boundaries, $\alpha = 20°$, $p = 2$, $q = 2$

	$M_\infty = 0.05$		
Grids	No PC	Turkel PC	Block-Jacobi PC
1	5465	1145	2817
2	7381	564	1824
3	7697	315	1714
4	7777	308	1755
	$M_\infty = 0.1$		
Grids	No PC	Turkel PC	Block-Jacobi PC
1	2889	1169	1785
2	3984	575	1059
3	3427	315	1087
4	3497	308	1105
	$M_\infty = 0.2$		
Grids	No PC	Turkel PC	Block-Jacobi PC
1	1913	1169	1025
2	2116	575	687
3	1750	315	725
4	1780	308	737

The same matrix of tests was also performed with the imposition of periodic boundaries. The results of these first- and second-order tests are tabulated in Tables 3 and 4, respectively. The results listed as DNC* did not reach the desired level of convergence after 5000 multigrid cycles. As before, it is quite apparent that the preconditioned systems perform significantly better than the original system, with Turkel again performing best. With this periodic boundary condition, however, the advantage of the Turkel preconditioner's low-frequency error propagation characteristics are neutralized and only its dissipation characteristics can be deemed responsible for the improvement over the block-Jacobi preconditioner. This theory is further supported by the first-order $M_\infty = 0.2$ results, where the block-Jacobi preconditioner seems to begin "catching up" with the Turkel preconditioner. A Fourier footprint analysis shows that the damping properties of the two systems are even more comparable at this flow speed.

Finally, it is noted that several of the solutions are listed in Table 4 as unstable. The poor damping in these cases proved to be insufficient to eliminate the additional errors introduced by intergrid transfers during each multigrid cycle.

Table 3. Work Requirements for Convergence, First-Order Solver, 64x32 Square Grid, Periodic Boundaries, $\alpha = 20°$, $p = 2$, $q = 2$

	$M_\infty = 0.05$		
Grids	No PC	Turkel PC	Block-Jacobi PC
1	DNC*	5669	5985
2	33951	3670	4026
3	17577	1289	2413
4	15986	724	2261
	$M_\infty = 0.1$		
Grids	No PC	Turkel PC	Block-Jacobi PC
1	39145	6057	6097
2	20776	3832	4176
3	10792	1289	2917
4	9471	776	2935
	$M_\infty = 0.2$		
Grids	No PC	Turkel PC	Block-Jacobi PC
1	24433	7141	6409
2	13070	4476	4314
3	6732	1275	2835
4	5682	773	2401

Table 4. Work Requirements for Convergence, $\kappa = 0$ Solver, 64x32 Square Grid, Periodic Boundaries, $\alpha = 20°$, $p = 2$, $q = 2$

	$M_\infty = 0.05$		
Grids	No PC	Turkel PC	Block-Jacobi PC
1	DNC*	DNC*	DNC*
2	DNC*	32019	DNC*
3	unstable	4573	DNC*
4	unstable	2810	unstable
	$M_\infty = 0.1$		
Grids	No PC	Turkel PC	Block-Jacobi PC
1	DNC*	DNC*	DNC*
2	DNC*	21297	DNC*
3	unstable	4754	DNC*
4	unstable	2810	unstable
	$M_\infty = 0.2$		
Grids	No PC	Turkel PC	Block-Jacobi PC
1	DNC*	DNC*	DNC*
2	DNC*	36564	DNC*
3	unstable	4838	37467
4	unstable	2932	unstable
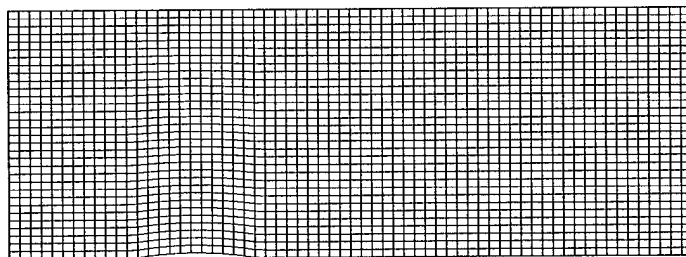
Fig. 8. 64x32 structured grid, solid circular arc bump, height $= 0.042c$.

Finally, flow over a solid bump was considered. The domain for this test was 5.5 chord lengths long and 2 chord lengths high; the height of the circular arc bump was 4.2% of the chord length. (This geometry was used by Lynn [5] and was incorporated here in order to verify correct code development.) A simple 64x32 unclustered algebraic grid was applied, as shown in Figure 8. The solid lower wall was simulated by the use of reflected ghost cells across the lower boundary. Free boundary conditions were applied to each of the other three boundaries using a ghost-cell approach.

As in the first two test sets, first- and second-order solvers were applied over three freestream Mach numbers, and 1-, 2-, 3-, and 4-level V-cycle multigrid algorithm were used. The work requirements for convergence in these cases are listed in Tables 5 and 6. As with the previous two test sets, preconditioning was beneficial in every case, but again the Turkel preconditioner provided the greater benefit. In this group of tests, three free boundaries again allowed error propagation to play a role in error removal; thus, the Turkel preconditioner's two-fold advantages of better high-frequency dissipation and low-frequency propagation can be credited with providing the best performance of the three systems.

Normalized residual histories for a representative sample of the second-order tests are also plotted. Figures 9-11 compare the effectiveness of each preconditioner in accelerating convergence for single-grid and 4-level multigrid algorithms. These plots again show that, while the block-Jacobi preconditioner provides definite improvement over the system without preconditioning, the Turkel preconditioned system invariably allows faster convergence than the block-Jacobi preconditioned system.

Table 5. Work Requirements for Convergence, First-Order Solver, 64x32 Grid, Solid Bump, $t = 0.042c$, $p = 2$, $q = 2$

	$M_\infty = 0.05$		
Grids	No PC	Turkel PC	Block-Jacobi PC
1	9253	1221	4181
2	9357	739	4232
3	10063	730	4511
4	10166	738	4548
	$M_\infty = 0.1$		
Grids	No PC	Turkel PC	Block-Jacobi PC
1	5469	1225	2437
2	5251	745	2626
3	5485	750	2733
4	5501	759	2713
	$M_\infty = 0.2$		
Grids	No PC	Turkel PC	Block-Jacobi PC
1	2801	1241	1241
2	3339	757	1326
3	3973	778	1384
4	4173	794	1447

Table 6. Work Requirements for Convergence, $\kappa = 0$ Solver, 64x32 Grid, Solid Bump, $t = 0.042c$, $p = 2$, $q = 2$

	$M_\infty = 0.05$		
Grids	No PC	Turkel PC	Block-Jacobi PC
1	10249	1897	5257
2	13434	912	5165
3	14874	749	5839
4	15284	1620	5999
	$M_\infty = 0.1$		
Grids	No PC	Turkel PC	Block-Jacobi PC
1	6345	1929	4145
2	8315	924	3309
3	8951	761	3584
4	9139	774	3668
	$M_\infty = 0.2$		
Grids	No PC	Turkel PC	Block-Jacobi PC
1	4193	2017	2737
2	3714	946	1531
3	3861	785	1509
4	4036	798	1559
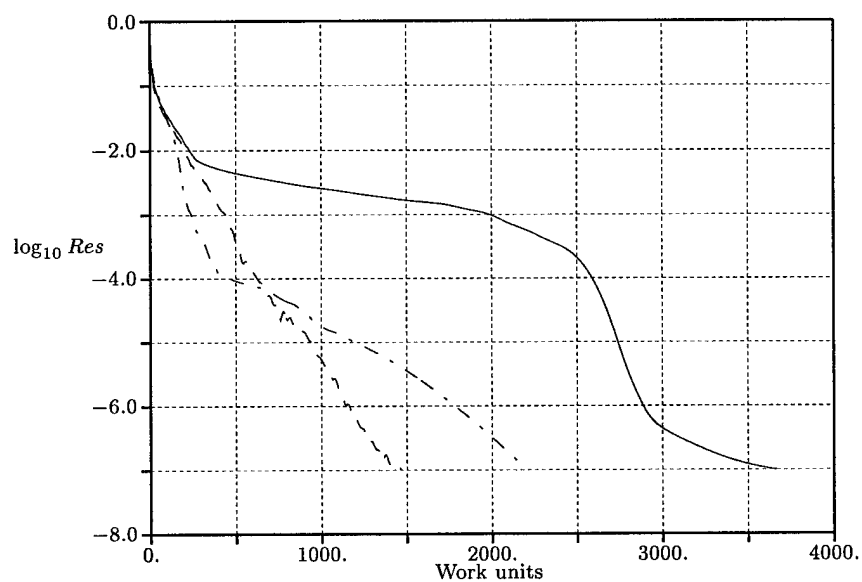
(a)
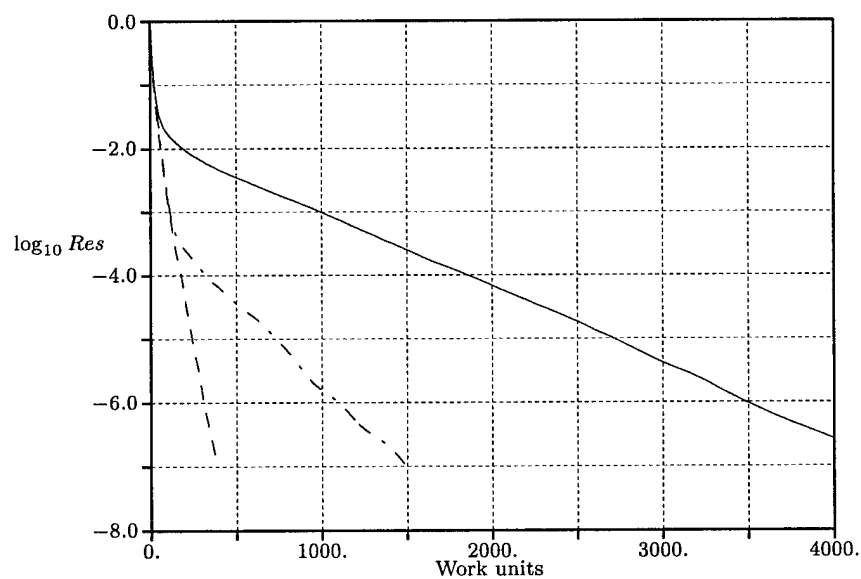
(b)

Fig. 9. Preconditioner effectiveness: free boundaries, 64x32 square grid, $\alpha = 20°$, $M_\infty = 0.1$, second-order solver, $p = 2$, $q = 2$. Solid: Euler; Dashed: Turkel; Dash-Dot: Block-Jacobi. a) 1 grid level and b) 4 grid levels.
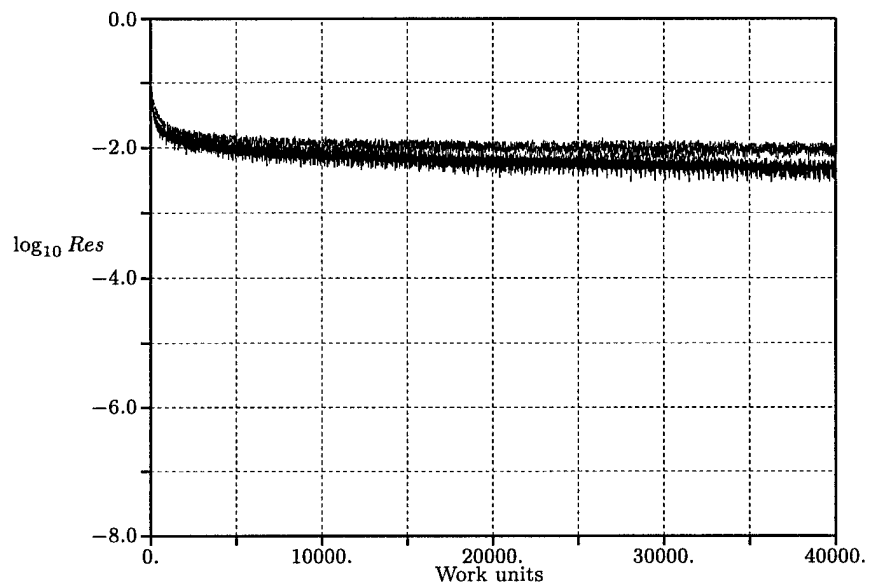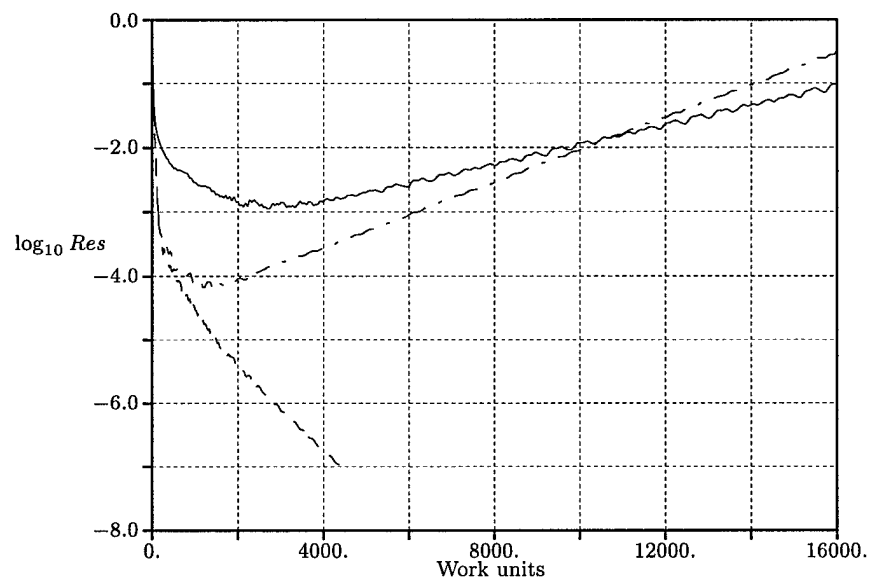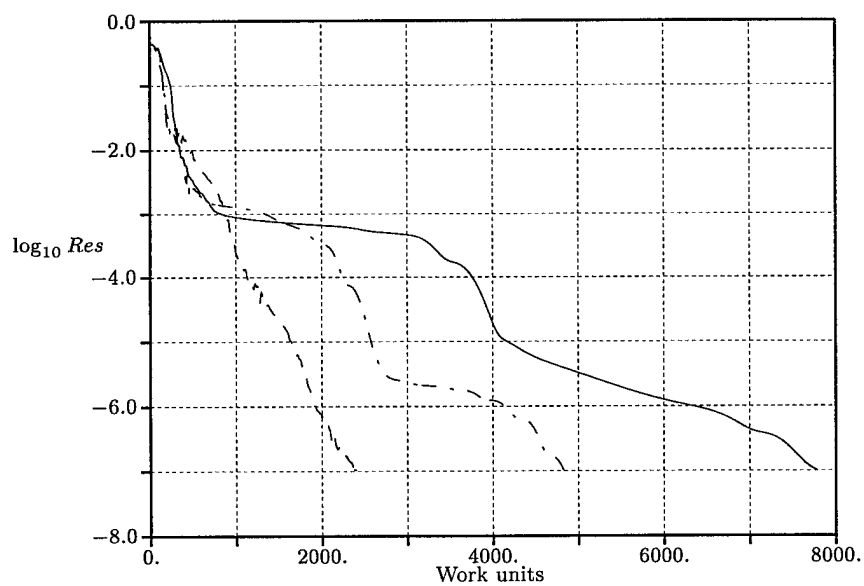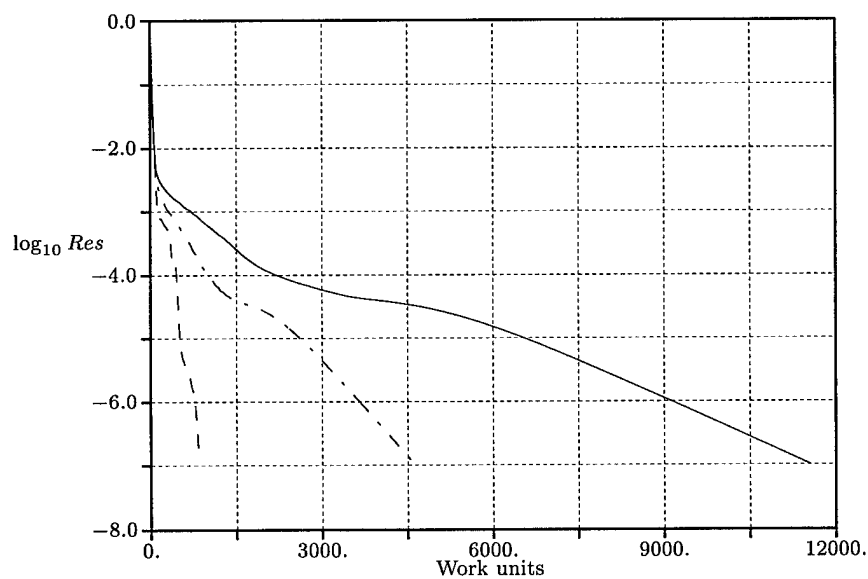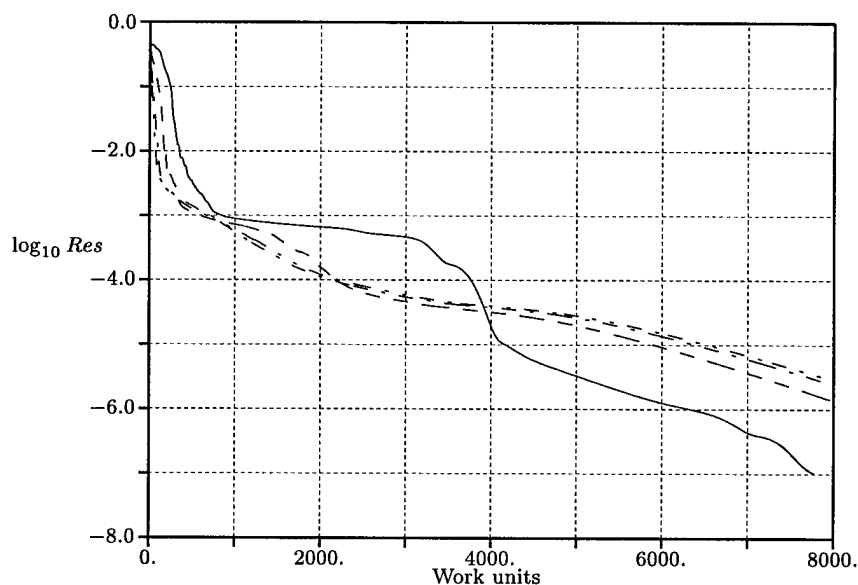
Fig. 10. Preconditioner effectiveness: periodic boundaries, 64x32 square grid, $\alpha = 20°$, $M_\infty = 0.1$, second-order solver, $p = 2$, $q = 2$. Solid: Euler; Dashed: Turkel; Dash-Dot: Block-Jacobi. a) 1 grid level and b) 4 grid levels.

Fig. 11. Preconditioner effectiveness: solid bump, 64x32 grid, $M_\infty = 0.1$, second-order solver, $p = 2$, $q = 2$. Solid: Euler; Dashed: Turkel; Dash-Dot: Block-Jacobi. a) 1 grid level and b) 4 grid levels.

Fig. 12. Multigrid effectiveness: no preconditioner, 64x32 grid, solid bump, $M_\infty = 0.1$, second-order solver, $p = 2$, $q = 2$. Solid: 1 grid; Dashed: 2 grids; Dash-Dot: 3 grids; Dash-Dot-Dot: 4 grids.

B. Multigrid Effectiveness

A second conclusion can be drawn from these results. Convergence of the two preconditioned systems is generally accelerated by utilizing more multigrid levels, with some exceptions due to the fact that each ever-coarser grid adds additional computational work while acting upon fewer and fewer error components. Conversely, the use of multigrid with the unpreconditioned system often *slowed* convergence, in several cases dramatically. Some examples of this phenomenon are illustrated in Figures 12-14. It becomes apparent that when preconditioning is not used to enhance the high-frequency damping properties of a solver, the benefits of multigrid may not be realized.
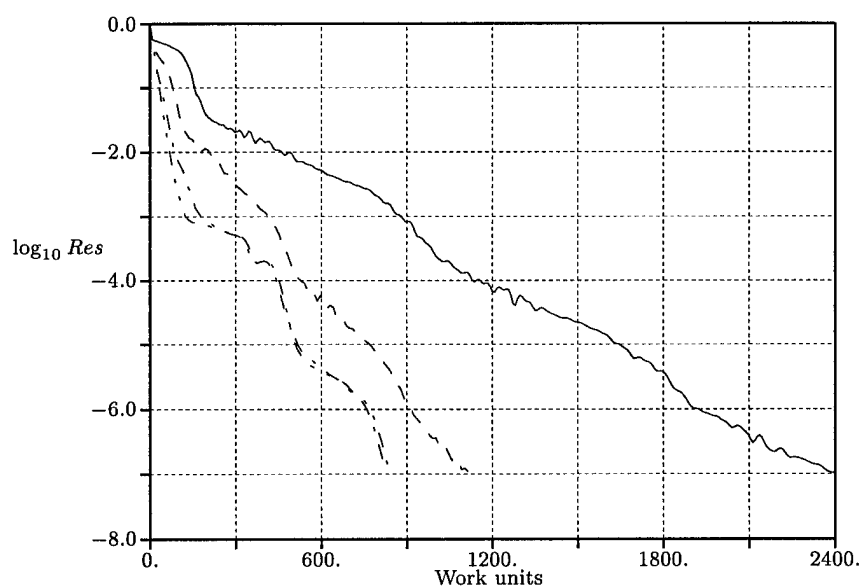
Fig. 13. Multigrid effectiveness: Turkel preconditioner, 64x32 grid, solid bump, $M_\infty = 0.1$, second-order solver, $p = 2$, $q = 2$. Solid: 1 grid; Dashed: 2 grids; Dash-Dot: 3 grids; Dash-Dot-Dot: 4 grids.
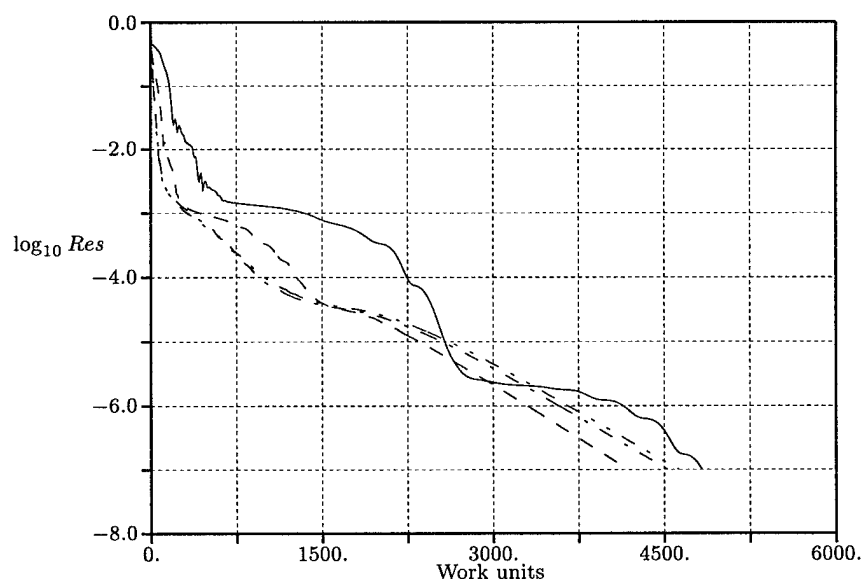

Fig. 14. Multigrid effectiveness: block-Jacobi preconditioner, 64x32 grid, solid bump, $M_\infty = 0.1$, second-order solver, $p = 2$, $q = 2$. Solid: 1 grid; Dashed: 2 grids; Dash-Dot: 3 grids; Dash-Dot-Dot: 4 grids.

C. Mach Number Dependence

A third conclusion is drawn from these results. The tables show that, in general, convergence rates for the unpreconditioned and block-Jacobi preconditioned systems improve as Mach number increases. This is due to the fact that high-frequency error modes lying close to the $g = 1$ contour at very low freestream Mach numbers gradually pull away and are damped more efficiently at higher Mach numbers. The Turkel preconditioned system, however, exhibits a distinctly different trend. In fact, convergence is essentially independent of Mach number. These trends are illustrated here in Figures 15-18. This interesting result indicates that, at very low Mach numbers, block-Jacobi performance will deteriorate while the Turkel preconditioner will continue to accelerate convergence, making it the preconditioner of choice at very low flow speeds. (It is noted that the single unusual residual history in Figure 18b, which was the only case to exhibit this unexpected behavior, remains unexplained.)
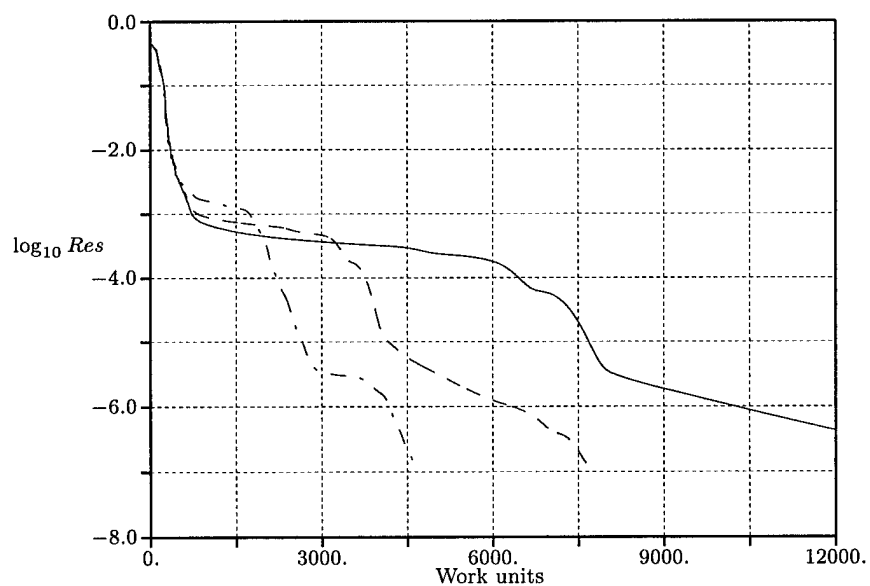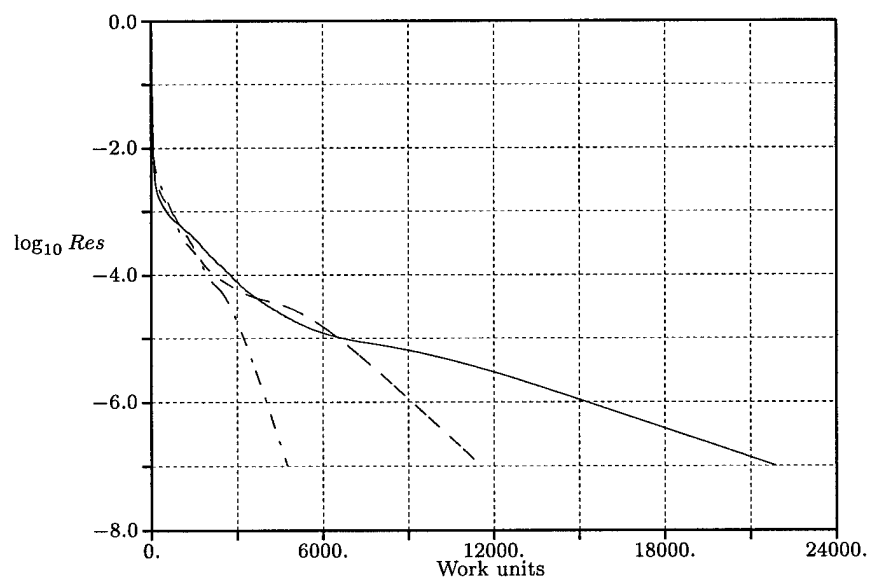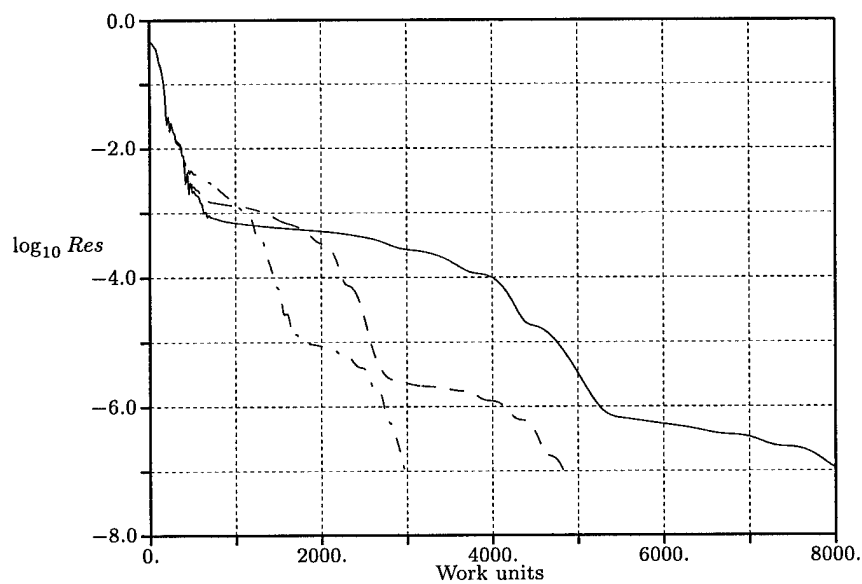
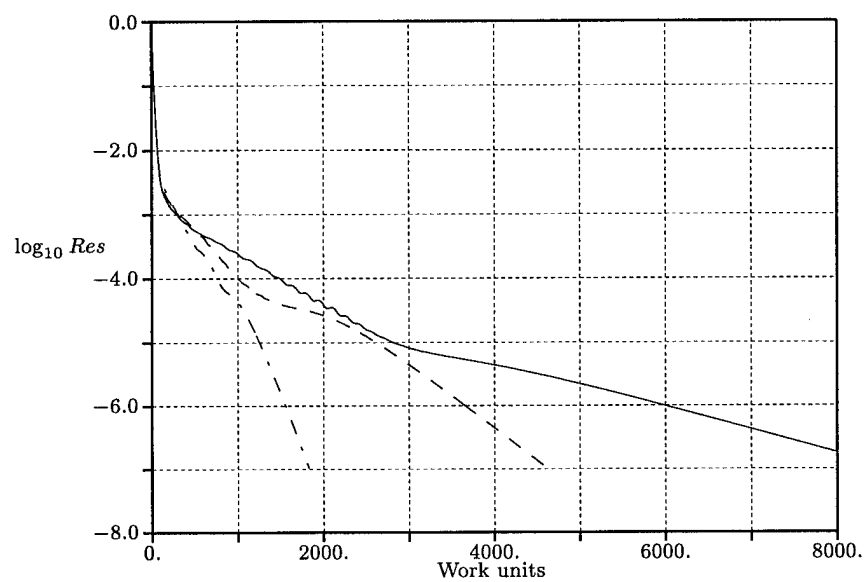Fig. 15. Mach number dependence: no preconditioner, 64x32 grid, solid bump, second-order solver, $p = 2$, $q = 2$. Solid: $M_\infty = 0.05$; Dashed: $M_\infty = 0.1$; Dash-Dot: $M_\infty = 0.2$. a) 1 grid level and b) 4 grid levels.

Fig. 16. Mach number dependence: block-Jacobi preconditioner, 64x32 grid, solid bump, second-order solver, $p = 2$, $q = 2$. Solid: $M_\infty = 0.05$; Dashed: $M_\infty = 0.1$; Dash-Dot: $M_\infty = 0.2$. a) 1 grid level and b) 4 grid levels.
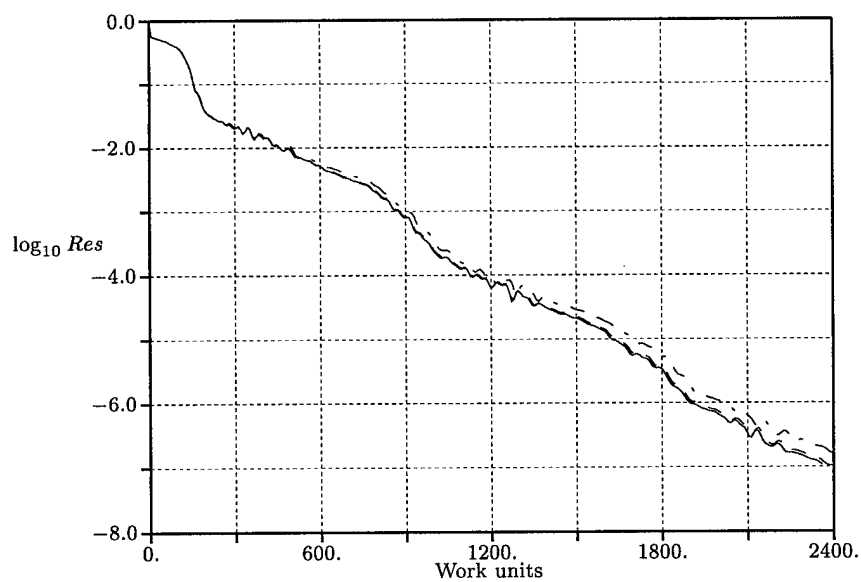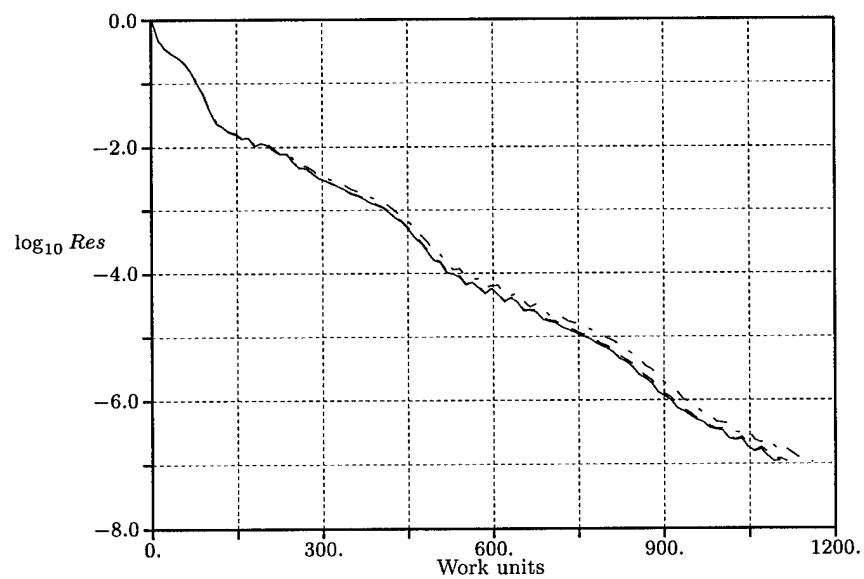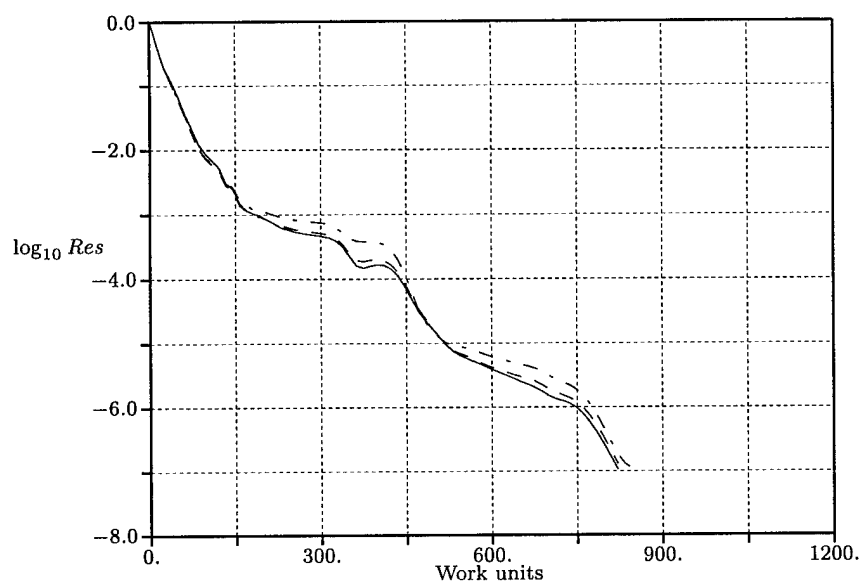
(a)

(b)

Fig. 17. Mach number dependence: Turkel preconditioner, 64x32 grid, solid bump, second-order solver, $p = 2$, $q = 2$. Solid: $M_\infty = 0.05$; Dashed: $M_\infty = 0.1$; Dash-Dot: $M_\infty = 0.2$. a) 1 grid level and b) 2 grid levels.
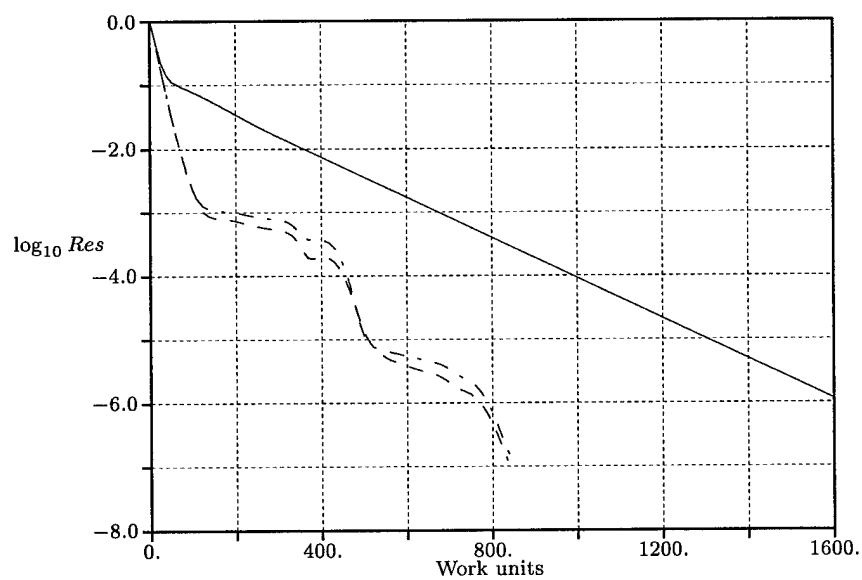
Fig. 18. Mach number dependence: Turkel preconditioner, 64x32 grid, solid bump, second-order solver, $p = 2$, $q = 2$. Solid: $M_\infty = 0.05$; Dashed: $M_\infty = 0.1$; Dash-Dot: $M_\infty = 0.2$. a) 3 grid levels and b) 4 grid levels.

D. Solution Accuracy

A final observation is made based on the results of the solid bump flow tests. Figures 19-20 show the second-order accurate converged solutions for the original system and the two preconditioned systems with $M_\infty = 0.05$. In Figure 19, total pressure contours are plotted; the distribution of total pressure in an inviscid flow should remain constant everywhere in the flow. Unexpectedly, the solution to the unpreconditioned and block-Jacobi preconditioned systems shows a small but noticeable variation in total pressure. In contrast, the Turkel preconditioned solution shows almost no variation in total pressure, clearly making it a more accurate solution. Figure 20 shows the Mach number distributions of the converged solutions. Again, the solutions for the Turkel-preconditioned system are noticeably different from the other two solutions. In fact, solution to the Turkel preconditioned system has more symmetric Mach contours about the bump, thus indicating a decreased presence of numerical dissipation as a result of the modification to the Roe scheme. Thus, these results agree with the conclusions in [18, 19, 20] that the Turkel preconditioner improves the accuracy of a solution.
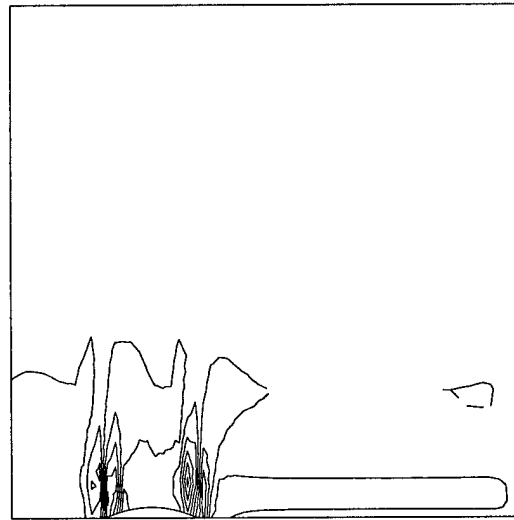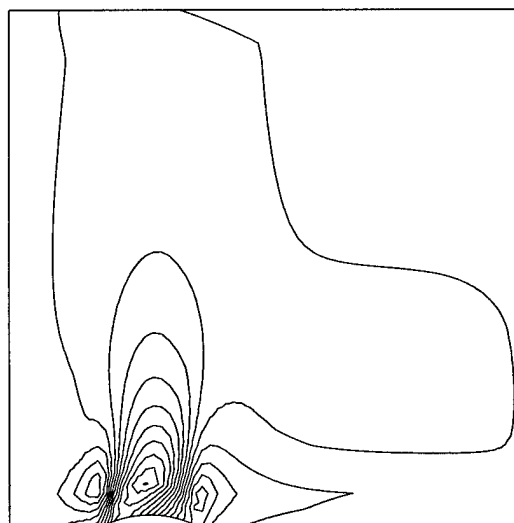
(a)
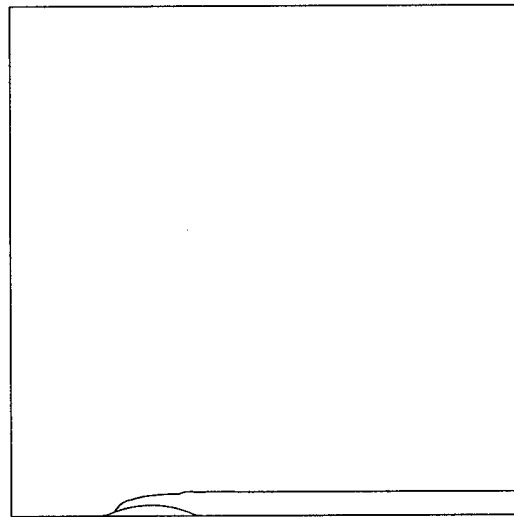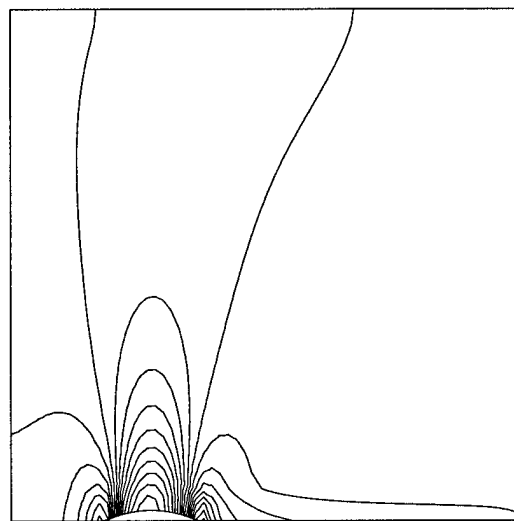

(b)

Fig. 19. Total pressure contours from 0.7153 to 0.7158 in 20 steps. Second-order solver, $M_\infty = 0.05$. a) No preconditioner / block-Jacobi preconditioner and b) Turkel preconditioner.

(a)

(b)

Fig. 20. Mach contours from 0.046 to 0.055 in 20 steps. Second-order solver, $M_\infty = 0.05$. a) No preconditioner / block-Jacobi preconditioner and b) Turkel preconditioner.

CHAPTER IV

CONCLUSIONS

In this study, several simple but effective tests were performed to evaluate and compare the utility of the Turkel and block-Jacobi preconditioners when employed in a multigrid algorithm to solve the compressible Euler equations. As expected, use of the multigrid algorithm generally served to accelerate convergence, and the two preconditioned systems converged significantly more quickly than the original system.

Work already done by others indicates that the two preconditioners have similar high-high frequency error-damping properties. However, it is recognized that the Turkel preconditioner has superior convective characteristics. As a result of this improved propagation, we have shown slightly improved damping for high-low and low-high modes which must be eliminated by an iterative scheme when applying full-coarsening multigrid. The most interesting question studied here, therefore, is whether the slightly better damping properties and significantly improved convective characteristics can have a significant influence on convergence rates. The first- and second-order results generated clearly indicate that they do indeed play an important role in accelerating convergence to a steady solution. In every test performed, the Turkel-preconditioned system converged more rapidly than the block-Jacobi preconditioned system. Thus, based on the results of the work, it is concluded that there exist better local preconditioners than block-Jacobi for the two-dimensional compressible Euler equations, especially at low freestream Mach numbers.

Other observations were made about the systems tested. First, visualization of the flowfield over the solid bump showed that the Turkel-preconditioned system provided more accurate results than did the unpreconditioned and block-Jacobi preconditioned systems. Second, it was seen that the convergence rates for the Turkel-

preconditioned system were essentially independent of freestream Mach number at these low speeds. Finally, it was observed that, particularly for second-order testing, the use of multigrid without preconditioning generally *slowed* convergence rates. Thus, it is concluded that, for higher-order CFD applications, multigrid must be used in conjuction with local preconditioning in order to be effective.

Future studies are recommended to further validate the conclusions drawn here. Use of a semi-coarsening algorithm would distinctly delineate the relative impact of the Turkel preconditioner's damping and error-propagation properties on convergence acceleration. Repeated studies over a wider range of Mach numbers would clarify the extent of the validity of these conclusions over the subsonic flow regime. (It is hypothesized that the Turkel preconditioner will continue to outperform the block-Jacobi preconditioner at lower Mach numbers, whereas the two preconditioners will begin to perform more similarly at higher Mach numbers.) Grid dependence studies should be performed before these conclusions can be extended to significantly finer grids. Airfoil simulations would extend the results to another important engineering application. Finally, simulations should be performed with other continuum-based local precondtioners, such as that developed by van Leer, in order to draw broad conclusions about the effectiveness of this entire class of preconditioners.

REFERENCES

[1] A. Brandt. Guide to multigrid development. *Multigrid Methods, Lecture Notes in Mathematics*, 960, 1982.

[2] P. Wesseling. *An Introduction to Multigrid Methods*. John Wiley & Sons, 1992.

[3] S. Allmaras. Analysis of a local matrix preconditioner for the 2-D Navier-Stokes equations. AIAA Paper 93-3330-CP, 1993.

[4] S. Allmaras. Analysis of semi-implicit preconditioners for multigrid solution of the 2-D compressible Navier-Stokes equations. AIAA Paper 95-1651, 1995.

[5] J. F. Lynn. *Multigrid Solution of the Euler Equations with Local Preconditioning*. PhD thesis, University of Michigan, 1995.

[6] J. F. Lynn and B. van Leer. Multistage schemes for the Euler and Navier-Stokes equations with optimal smoothing. AIAA Paper 93-3355, 1993.

[7] N. A. Pierce and M. B. Giles. Preconditioning on stretched meshes. Technical Report 95/10, Oxford University Computing Laboratory, June 1995.

[8] C. H. Tai. *Acceleration Techniques for Explicit Codes*. PhD thesis, University of Michigan, 1990.

[9] P. Lotstedt. Grid independent convergence of the multigrid method for first-order equations. *SIAM Journal of Numerical Analysis*, 29(5):1370–1394, October 1992.

[10] B. van Leer, C.-H. Lee, and P. L. Roe. Characteristic time-stepping or local preconditioning of the Euler equations. AIAA Paper 91-1552, 1991.

[11] B. van Leer, C. H. Tai, and K. G. Powell. Design of optimally smoothing multi-stage schemes for the Euler equations. AIAA Paper 89-1933-CP, 1989.

[12] W. T. Lee. *Local Preconditioning of the Euler Equations*. PhD thesis, University of Michigan, 1991.

[13] D. Anderson, J. Tannehill, and R. Pletcher. *Computational Fluid Mechanics and Heat Transfer*. Taylor & Francis, 1984.

[14] B. van Leer. Upwind-difference mehods for aerodynamic problems governed by the Euler equations. *Large-Scale Computations in Fluid Mechanics, Lectures in Applied Mathematics*, 22, 1985.

[15] P. L. Roe. Approximate Riemann solvers, parameter vectors and difference schemes. *Journal of Computational Physics*, 43, 1981.

[16] W. A. Mulder. A new multigrid approach to convection problems. *Journal of Computational Physics*, 83:303–323, 1992.

[17] D. L. Darmofal and P. J. Schmid. The importance of eigenvectors for local preconditioners of the Euler equations. AIAA Paper 95-1655, 1995.

[18] A. G. Godfrey. *Topics on Spatially High-Order Accurate Methods and Preconditioning for the Navier-Stokes Equations with Finite-Rate Chemistry*. PhD thesis, Virginia Polytechnic Institute and State University, December 1992.

[19] C. L. Reed. *Low Speed Preconditioning Applied to the Compressible Navier-Stokes Equations*. PhD thesis, University of Texas at Arlington, 1995.

[20] E. Turkel, A. Fiterman, and B. van Leer. Preconditioning and the limit to the incompressible flow equations. Technical Report 93-42, ICASE, 1993.

[21] E. Turkel. Preconditioned methods for solving the incompressible and low speed compressible equations. *Journal of Computational Physics*, 72:277–298, 1987.

[22] E. Turkel. Review of preconditioning methods for fluid dynamics. Technical Report 92-47, ICASE, 1992.

VITA

Barrett Taylor McCann, of San Antonio, Texas, is currently a second lieutenant in the United States Air Force. In May 1995, he earned the degree of bachelor of science in aeronautical engineering at the United States Air Force Academy, where he received a National Science Foundation fellowship to continue his studies at Texas A&M University. His next assignment takes him to Eglin Air Force Base, near Fort Walton Beach, Florida. He is married to the former P. Suzann Klaver, of Denver, Colorado. He can be reached care of:

Bruce and Sandy Wilhite

7802 Misty Park

San Antonio, TX 78250